Simulation Tools for X-ray Surveyor

John ZuHone, Science Support Office

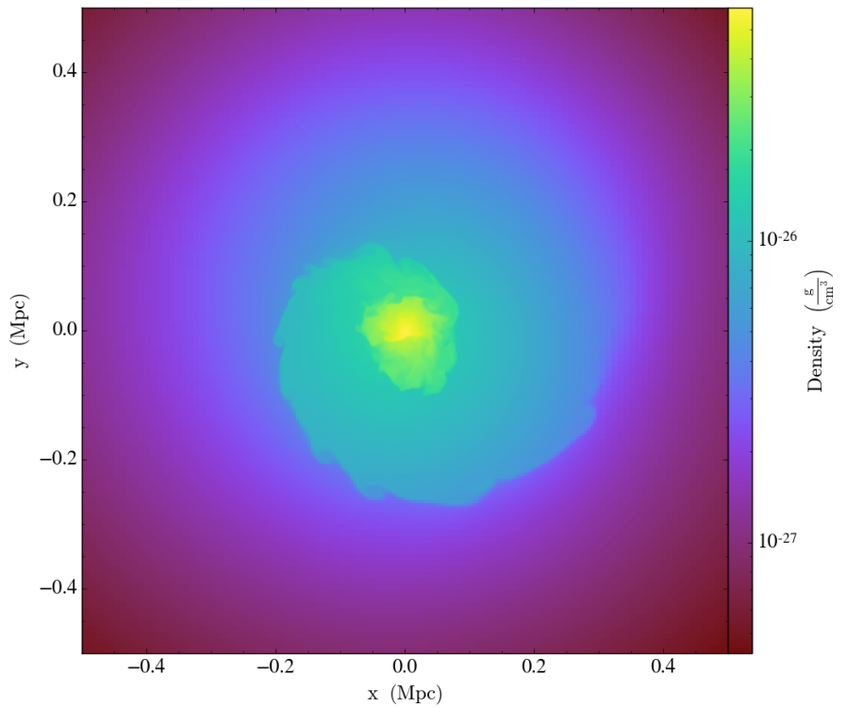
with

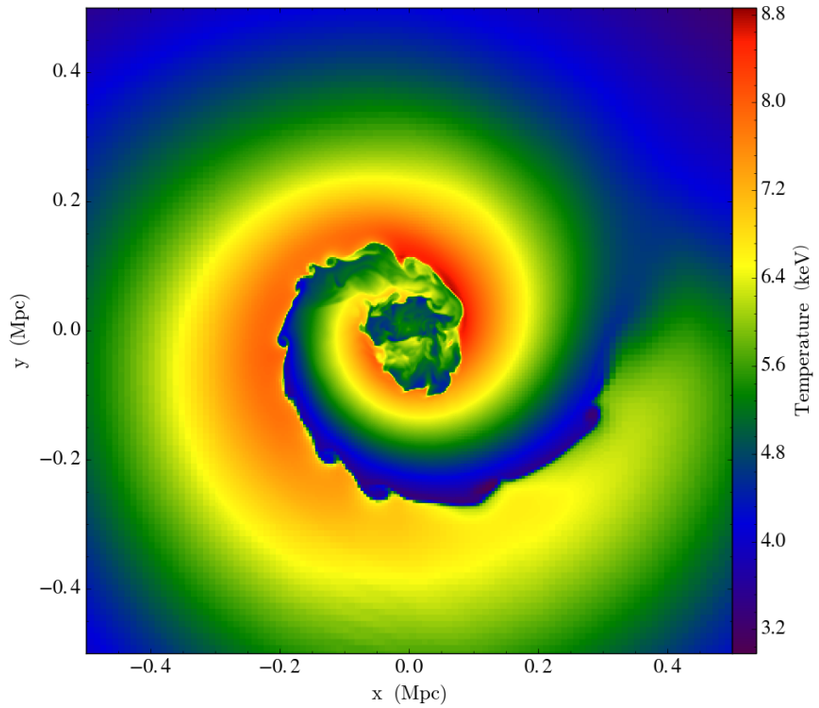
Alexey Vikhlinin, Scott Randall, Reinout Van Weeren,
Felipe Santos, Herve Bourdin, Ralph Kraft,
Akos Bogdan, Valsamo Antoniou, Francesca Civano

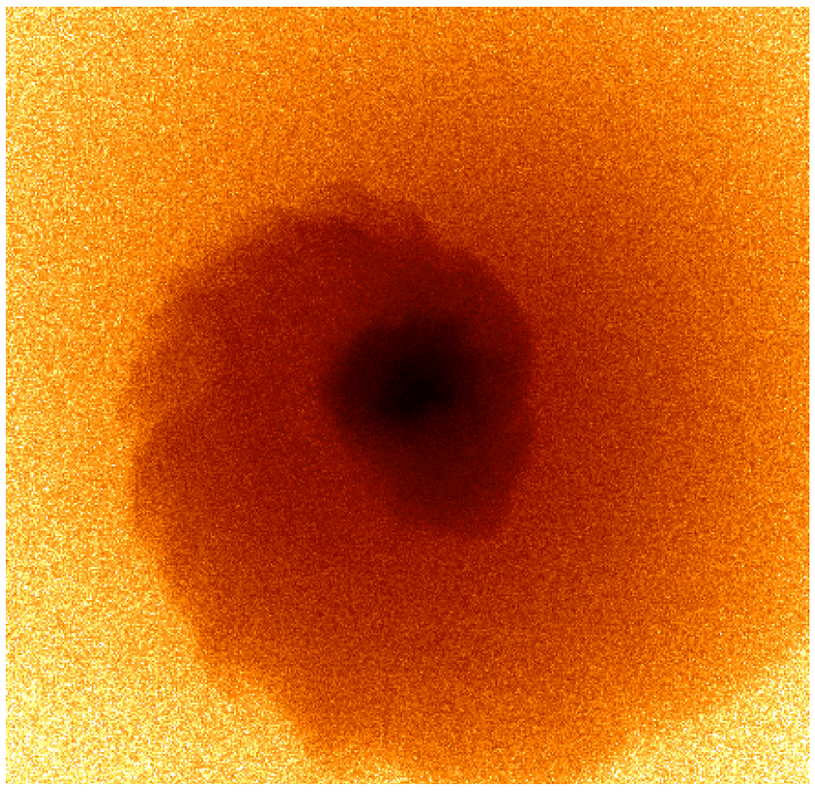
Science Support Office

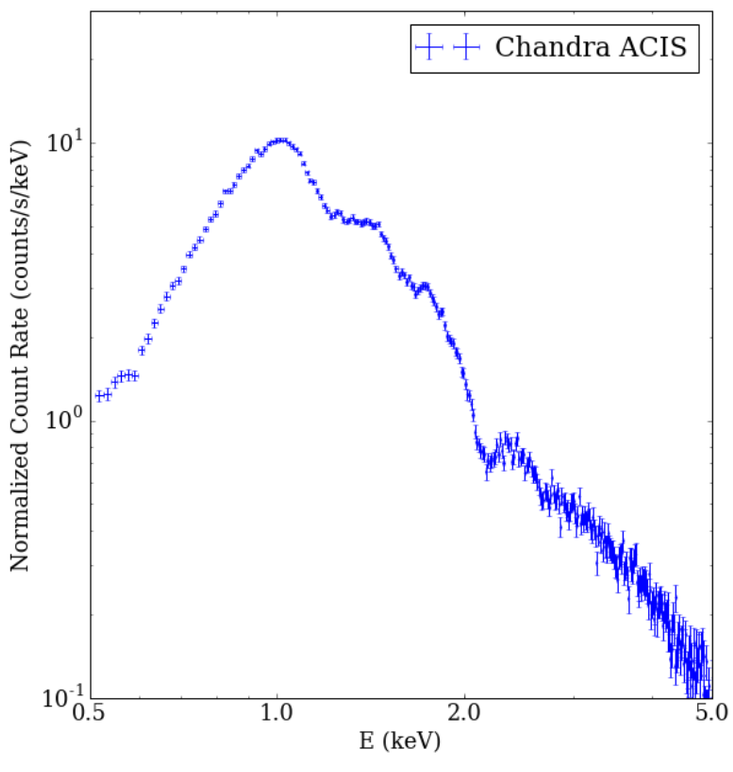
- What are we doing?
- Creating software tools to enable simulations of X-ray Surveyor observations
- Basic Philosophy:
  - Multiple points of entry
  - Support different types of simulations
  - Interoperability with other tools
  - Produce standard data products

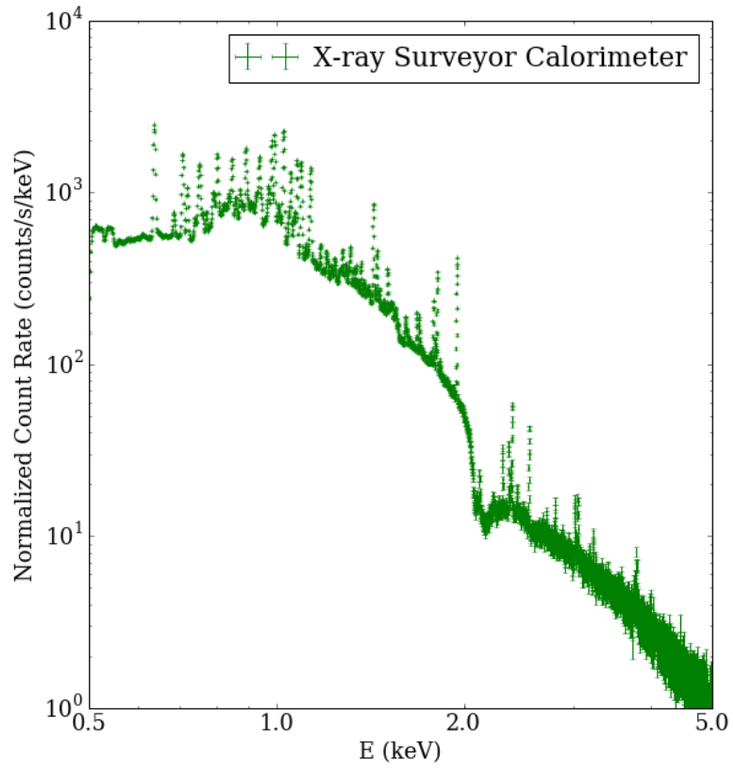
We've been hard at work!


0.4
0.2
0.0
−0.2
−0.4
y (Mpc)
−0.4
−0.2
0.0
0.2
0.4
x (Mpc)
10^{-26}
10^{-27}
Density $\left(\frac{g}{cm^3}\right)$


0.4
0.2
0.0
−0.2
−0.4
y (Mpc)
−0.4
−0.2
0.0
0.2
0.4
x (Mpc)
8.8
8.0
7.2
6.4
5.6
4.8
4.0
3.2
Temperature (keV)

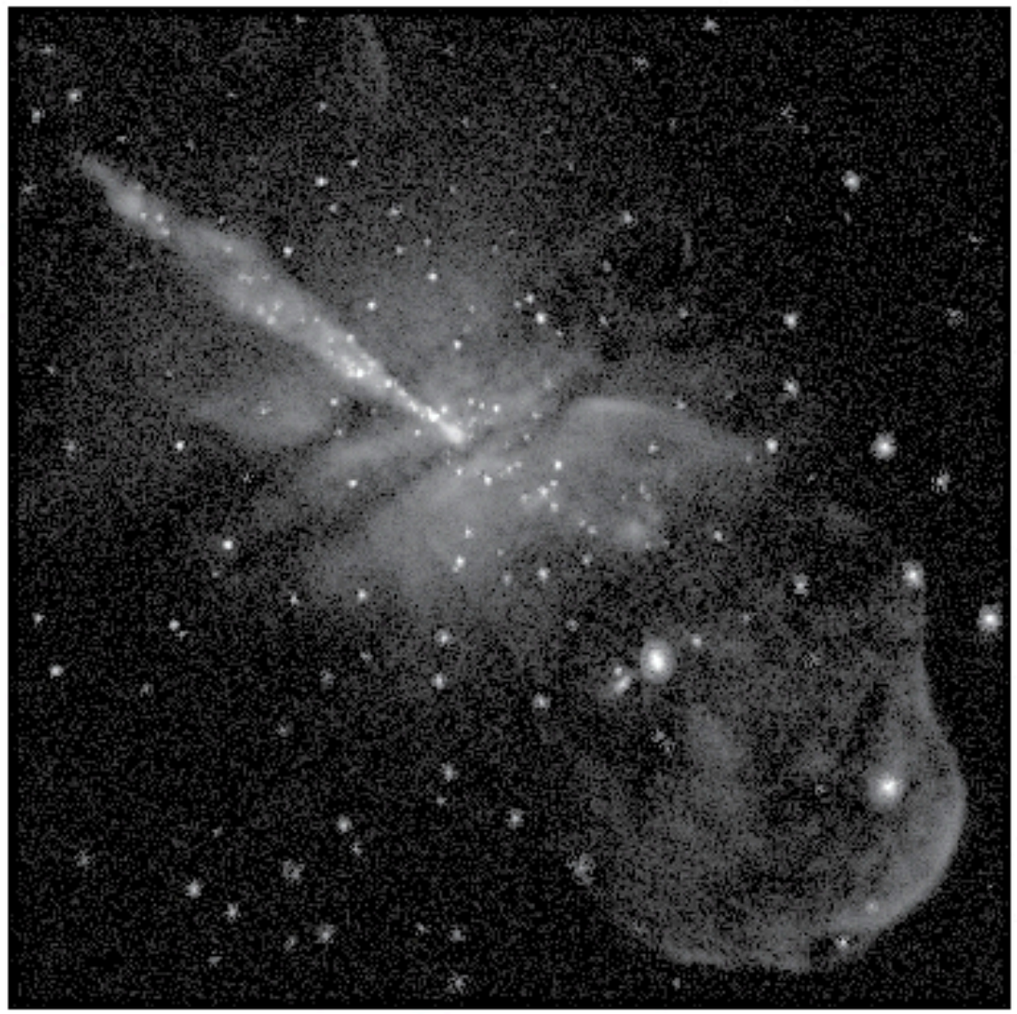



Chandra ACIS
10^{1}
10^{0}
10^{-1}
Normalized Count Rate (counts/s/keV)
0.5
1.0
2.0
5.0
E (keV)


X-ray Surveyor Calorimeter
10^{4}
10^{3}
10^{2}
10^{1}
10^{0}
Normalized Count Rate (counts/s/keV)
0.5
1.0
2.0
5.0
E (keV)

AXAF

XRS



Outline

- Three tools:
  - pyXSIM (ready for use)
  - XRStools (in development, ready next week)
  - XMAP Feature Extrapolation (in development)

pyXSIM /
3D Model

XRStools /
2D Model

XMAP
Feature
Extrapolation

SIMPUT
Catalog

XRStools
Instrument
Simulator:
PSF, ARF, RMF,
backgrounds,
etc.

“Standard”
Event
Products

pyXSIM /
3D Model

XRStools /
2D Model

XMAP
Feature
Extrapolation

SIMPUT
Catalog

XRStools
Instrument
Simulator:
PSF, ARF, RMF,
backgrounds,
etc.

for
comparisons…

MARX,
SIMX,
SIXTE

“Standard”
Event
Products

pyXSIM

- Python package for generating synthetic X-ray observations from 3D models (simulation outputs or “toy” models)

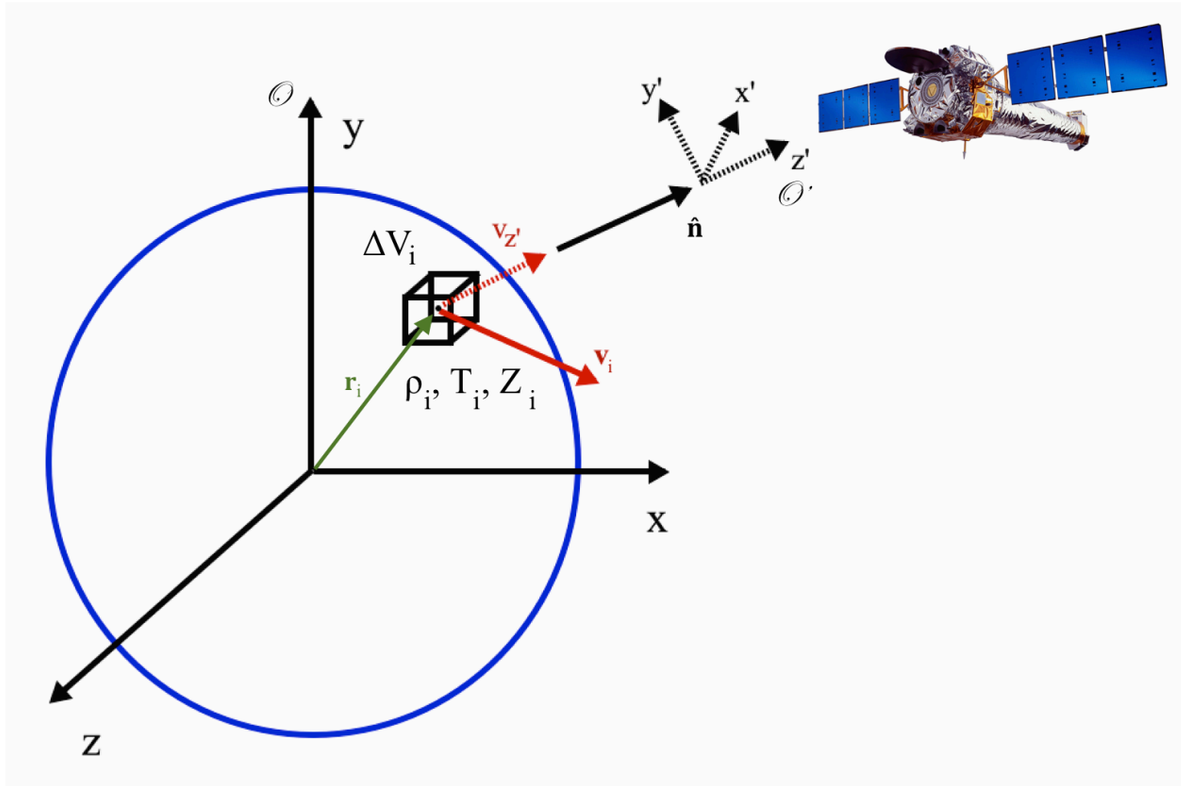
- Uses the yt Project (http://yt-project.org) to handle 3D data of grid or SPH type

pyXSIM

- pyXSIM assumes you have a 3D emission model of a source, from a simulation or perhaps just a 3D grid or collection of particles

- Assuming a large exposure time and/or collecting area, generate a large number of sample photons in the rest frame of the source

- Use this large sample to draw sub-sampled events which are projected onto the sky, Doppler and cosmologically shifted, and absorbed by Galactic foregrounds



pyXSIM

- Types of inputs: Anything yt can read, including simulation datasets (FLASH, Gadget, Athena, Enzo, etc.) or 3D NumPy arrays of grid points or particles

- Types of emission models:
    - Thermal models (e.g., APEC)
    - Power-law models
    - Line emission
    - Custom models you define

- pyXSIM also has its own built-in instrument simulator, but we'll be encouraging people to use the one from XRStools

pyXSIM

- Current version is 1.1.0, 1.1.1 to be released soon
- Installable via pip or Anaconda Python
- http://hea-www.cfa.harvard.edu/~jzuhone/pyxsim

XRStools

- An assortment of tools for simulating mock observations

    - Spectral and Spatial Models: create spectral models, generate simulated energies, create spatial models

    - SIMPUT I/O: Read and write models for emission from sources

    - Instrument Simulator: Convolve source models with XRS instrumental responses to produce mock observations

Spectral Models

- These models generate photon energies
- Spectral models:
    - Thermal (using AtomDB tables)
    - Power-law
    - From an XSPEC call
    - From a file
- Operations:
    - Add spectra together
    - Foreground absorption

Spatial Models

- These models generate photon coordinates
- Spatial Models:
  - Point sources
  - β-models
  - Model from a Python function
  - Model from a file

SIMPUT Catalogs

- SIMPUT == “SIMulated inPUT”
- Becoming a standard file format for mock X-ray observations
- http://www.sternwarte.uni-erlangen.de/research/sixte/simput.php
- We use the “photon list” specification: files contain lists of source RA, Dec, energy
- Enables interoperability with other tools: SIMX, SIXTE, MARX

Instrument Simulator

- Simulates response of an X-ray Surveyor instrument model:
  - Reads SIMPUT file
  - Uses effective area curve to determine which events are observed
  - Pixelizes events, applies spatial PSF and dithering
  - Creates spectral channels using RMF
  - Adds astrophysical and instrumental backgrounds

Instrument Simulator

- There are currently built-in models for the imager and the calorimeter

- However, it is easy to create your own model specification and use it as well

- Create a JSON file with the following information (for example):

```
{'name': 'hdxi', # The short name of the instrument
 'arf': 'xrs_hdxi.arf', # The file containing the ARF
 'rmf': 'xrs_hdxi.rmf' # The file containing the RMF
 'dtheta': 0.3333333333, # The central pixel scale in arcsec
 'num_pixels': 4096, # The number of pixels on a side in the FOV
 'psf_fwhm': 0.5} # The FWHM of the PSF, in arcseconds}
```

XRStools

- Two entry points:
  - Command-line scripts: simpler to use, but limited in scope
  - Python interface: more powerful

XRStools Command-line Interface

- The command-line interface consists of several scripts:
  - `make_powerlaw_spectrum`: Creates a power-law spectrum and writes it to a file
  - `make_thermal_spectrum`: Creates a thermal spectrum and writes it to a file
  - `make_point_source`: Creates a SIMPUT catalog for a point source from a spectrum in a file
  - `make_event_file`: Convolves photons from a SIMPUT catalog with the instrument simulator

XRStools

- First version, v0.1.0, to be released next week
- Subsequent versions will be rolled out over the next few weeks which will include (beyond what was presented today):
  - Background point sources
  - Time-dependence

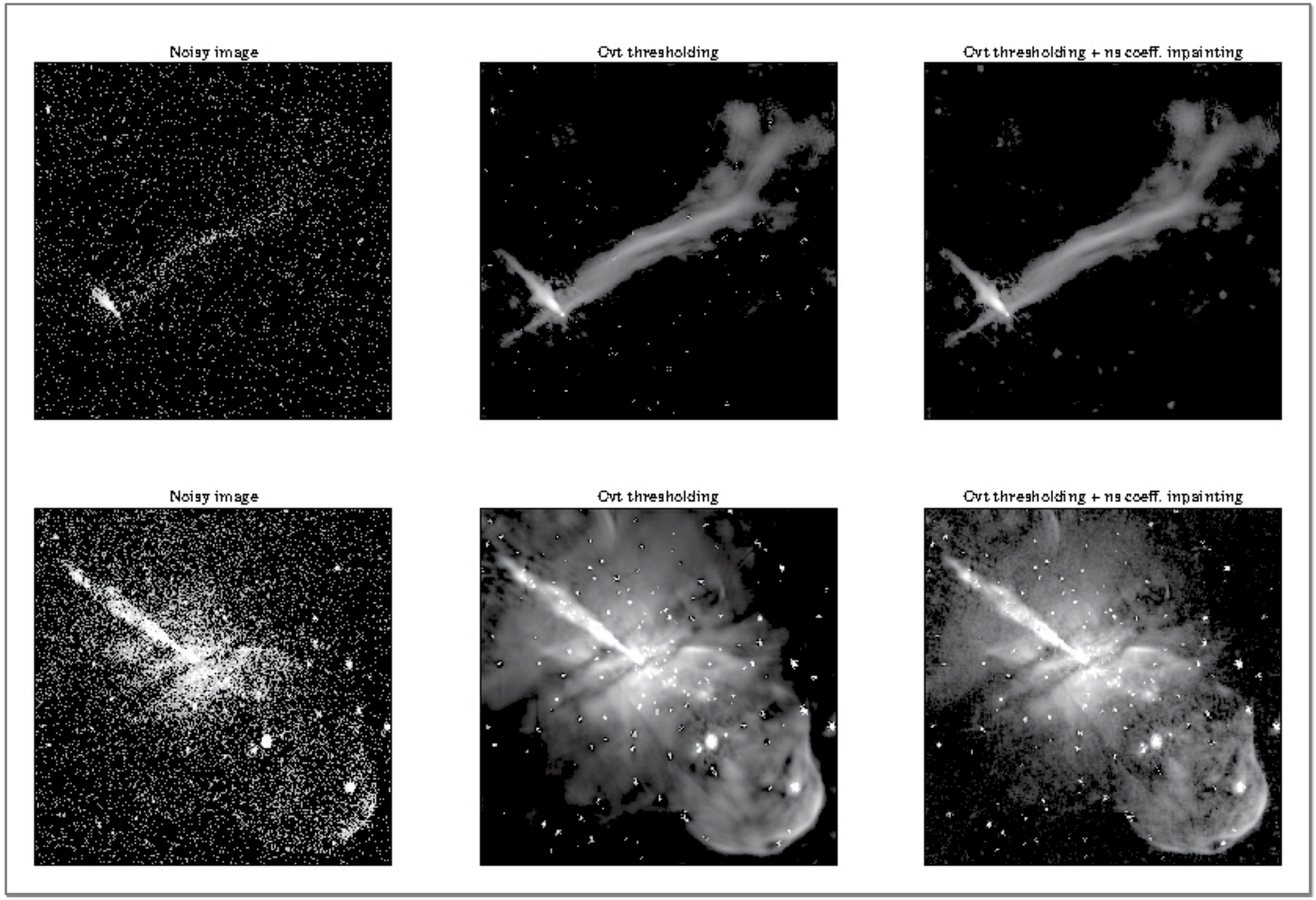
XRStools

- Development is being done on GitHub:
  - http://github.com/XRStools/xrs_tools

- Please feel free to clone the repository and make suggestions for enhancements or bug fixes (because there will be bugs)!

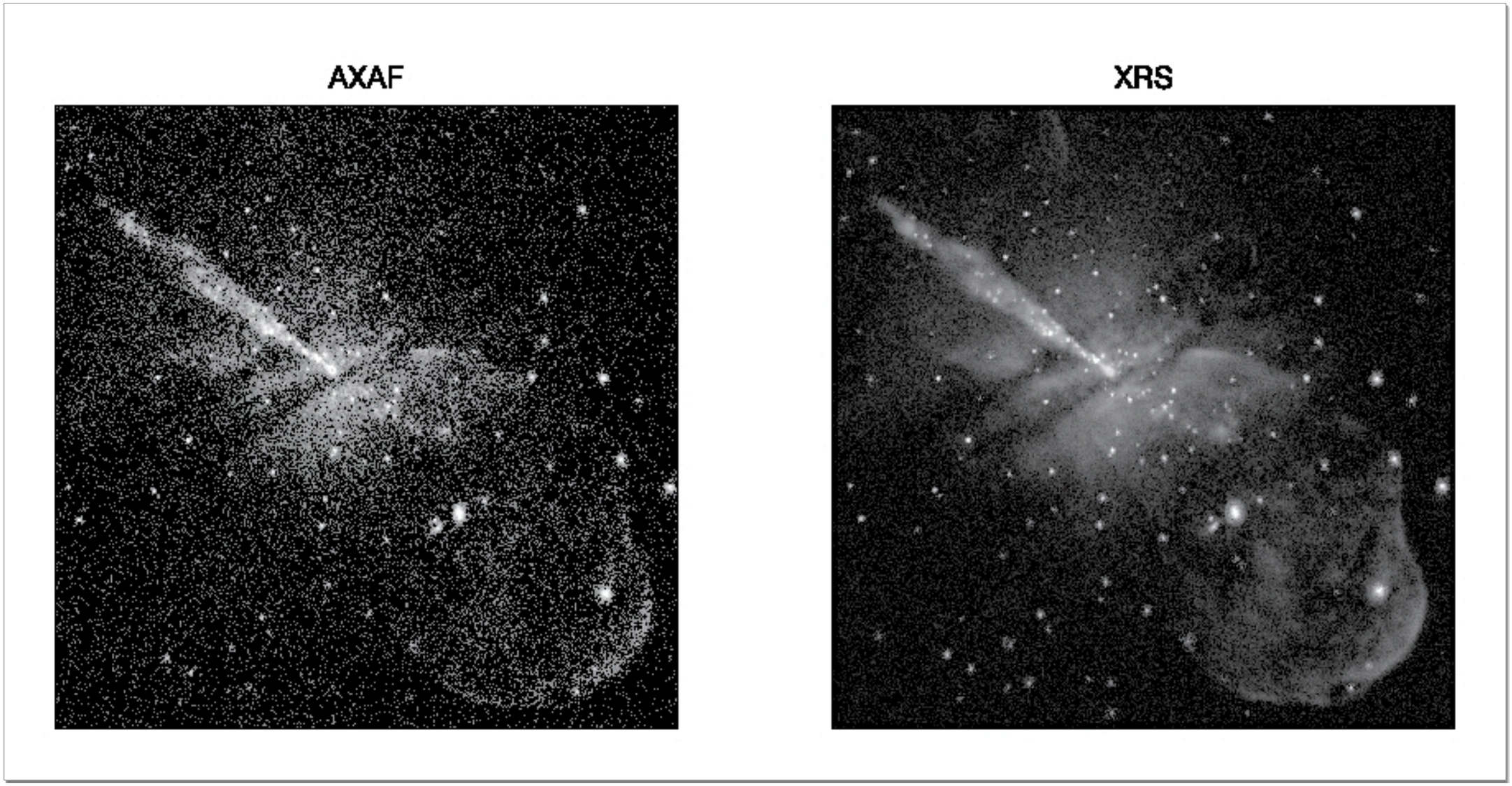
XMAP Feature Extrapolation

Feature extrapolation with curvelet transforms



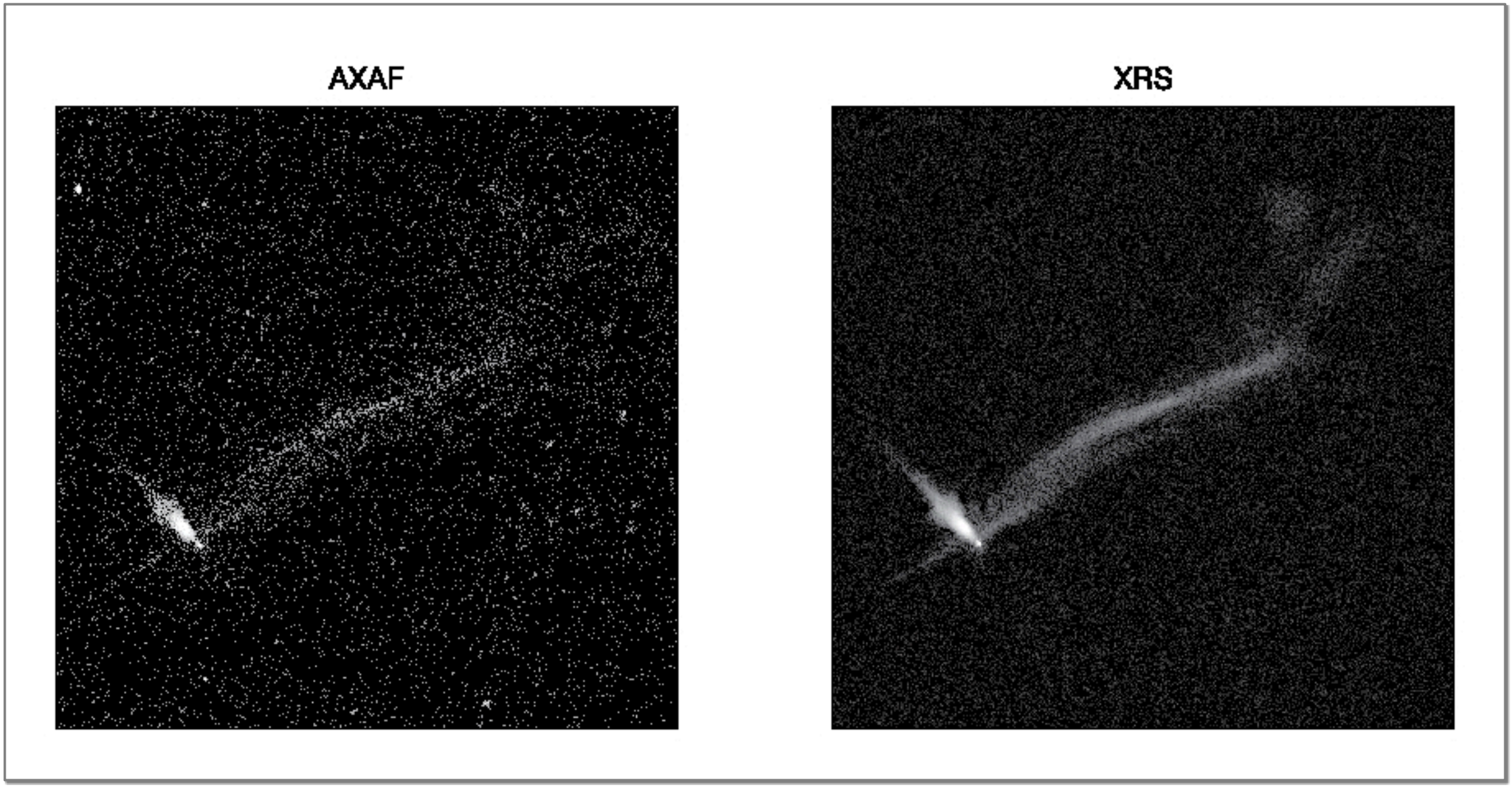
XMAP Feature Extrapolation

Mock X-ray Surveyor Observation of Centaurus A



XMAP Feature Extrapolation

Mock X-ray Surveyor observation of IGRJ11014-6103



Demos

Thank you!

Any questions?